

Implementación de Bootloaders en Microcontroladores PIC16 y PIC18 de Microchip Inc.

Ricardo Guadrón Gutiérrez.¹

Juan José Guevara Vásquez.²

Resumen

En este artículo se hace una descripción de las principales características de los programas bootloaders, los cuales se utilizan para programar microcontroladores PIC con una mínima cantidad de dispositivos electrónicos.

Palabras clave:

Microprocesadores, PIC, lenguajes de programación (computadores electrónicos), diseño de sistemas, microcontroladores.

Introducción

Los microcontroladores son parte fundamental de los equipos electrónicos tales como línea blanca, audio y video, así como dispositivos móviles. Esto los convierte en objeto de estudio obligatorio en institutos de formación técnica y universidades en áreas de la electrónica y sistemas embebidos. Al estudiar los microcontroladores y realizar el montaje de prototipos es indispensable pasar el firmware a la memoria de programa del microcontrolador por medio de un programador.

El programador es un equipo indispensable en laboratorios de experimentación debido a que la necesidad de programar varias veces al microcontrolador es muy alta; esto es por lo general un problema en la enseñanza de estos sistemas, ya que si bien una institución puede adquirir programadores, estos no siempre están disponibles para los estudiantes por limitaciones de espacio, tiempo y para no acortar la vida útil de los programadores existentes.

En sistemas microcontrolados, los bootloaders o cargadores de arranque son generalmente utilizados

para proveer un medio a través del cual se puedan borrar y escribir o programar datos en la memoria del microcontrolador. Esto trae consigo las siguientes ventajas:

1. Bajo costo de implementación.
2. No es necesario un programador externo³ para borrar y programar constantemente al microcontrolador.
3. Se facilita el diseño de placas de experimentación y desarrollo.
4. Códigos bootloader accesibles y gratuitos.

Las ventajas anteriores resultan inestimables en ambientes de enseñanza, ya que los estudiantes pueden construir un sistema de experimentación basado en microcontrolador y realizar, tanto prácticas como ejercicios, que les permiten solidificar sus conocimientos y competencias en la programación de microcontroladores.

El Programa Bootloader

Un bootloader es un pequeño programa que se almacena en una zona de memoria del microcontrolador y que por diseño se ejecuta al momento en que se inicializa al microcontrolador por medio de un reset.

Al inicializarse el microcontrolador, el vector de reset del bootloader (que se encuentra en la dirección 0000h), se encarga de redirigir la secuencia del programa al bootloader en la zona alta de la memoria⁴. Una vez que el cargador de arranque toma el control, verifica si se debe ingresar al “Modo Bootloader”. La orden de ingresar a este modo es externa y es generalmente originada por el usuario a través

(1) Ing. Electricista, Director Escuela de Ingeniería Eléctrica y Computación, ITCA-FEPADE, email: rguadron@itca.edu.sv

(2) Tec. en Ingeniería Eléctrica, Docente Escuela de Ingeniería Eléctrica, ITCA-FEPADE, email: juan.guevara@itca.edu.sv

(3) Se requiere de un programador externo únicamente cuando se graba el firmware del bootloader en la memoria del microcontrolador.

(4) La dirección de memoria cambia en función del tamaño de la memoria de programa del microcontrolador.

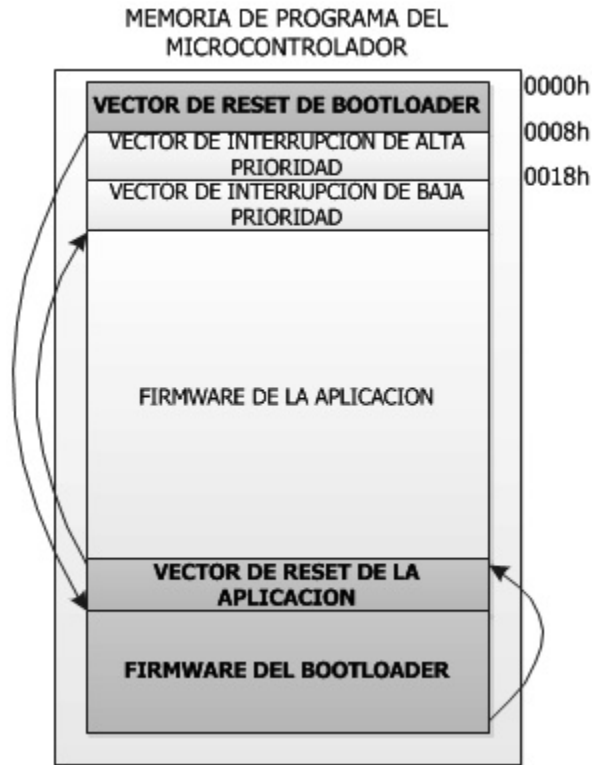


Fig. 1. Mapa de memoria del microcontrolador PIC16 en donde se puede apreciar la ubicación de los vectores de reset y firmware del bootloader. Las flechas indican la secuencia que el microcontrolador sigue al inicializarse.

de un software o por medio de una combinación de teclas, lo cual depende del tipo de bootloader que se está utilizando. Si se recibe la orden de ingresar al modo bootloader el programa entra en un bucle que le permite recibir órdenes de lectura, escritura y eliminación de datos (Tafanera, 2004).

Al finalizar el proceso de lectura-escritura o si no se recibe la orden de ingresar a modo bootloader, el programa pasa al vector de reset de la aplicación que redirige la secuencia del programa al firmware de la aplicación, el cual toma el control del microcontrolador hasta que se vuelva a producir una inicialización del sistema.

Actualmente, se pueden encontrar dos tipos de bootloader en función del puerto y protocolo que utilizan para comunicarse con el software de programación:

A) Bootloader Serie RS232

Utiliza el módulo USART de los microcontroladores para establecer comunicación con la computadora por medio de un puerto COM disponible. Este tipo de bootloader es ampliamente utilizado ya que casi todos los microcontroladores PIC16 y PIC18 poseen módulo USART. Cabe aclarar que tiene como limitante la necesidad de una circuitería de acople de señal y el hecho que los tradicionales puertos COM DB9 ya no se incluyen en las computadoras modernas, aunque esto puede solucionarse con la incorporación de un acople RS232-USB.

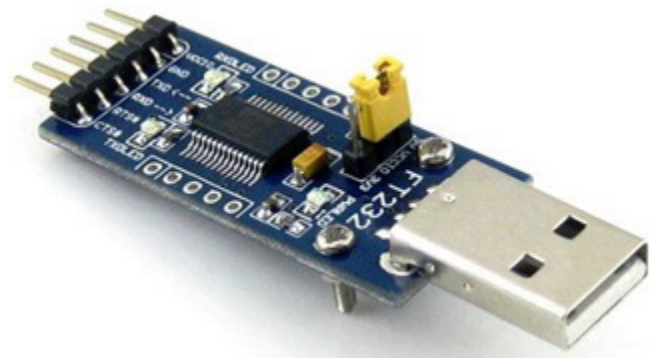


Fig. 2. Circuito de acople y conversión RS232 a USB que utiliza el chip FT232RL. Este circuito se instala como un puerto COM USB. En las computadoras que no disponen un puerto físico DB9, es la opción recomendada.

B) Bootloader Serie USB

Es utilizado por microcontroladores PIC de alta gama de 8 y de 16 bits que dispongan de un puerto de comunicaciones USB. Tiene la ventaja de no requerir circuitería de acople, además de proveer mayores velocidades para la lectura y escritura de datos.

Microchip Technology Inc. provee dos tipos de bootloader USB para sus microcontroladores: el primero es el MCHPFSUSB, el cual requiere de la instalación de controladores en el sistema operativo de la computadora en donde se conectará el microcontrolador; el otro tipo es HID, el cual no necesita la instalación de controladores (Valdés, 2007).

La nota de aplicación de Microchip AN1310 (Schlunder, 2010), provee un conjunto de recursos para la implementación de un bootloader de alta velocidad serie RS232 para microcontroladores PIC16 y PIC18.

Para implementar el bootloader en los PIC, se deben tomar en cuenta los siguientes requerimientos:

a) MPLAB IDE X

Es el IDE de Microchip basado en Netbeans con altas prestaciones para el desarrollador. Se instala junto con el compilador XC8, sucesor de C18.

b) Compilador MPASM v. 5.54

Es necesario para compilar correctamente el código del bootloader para PIC16, se instala junto con el compilador C18.

c) Compilador C18 v. 3.47

Se requiere para la compilación del código del bootloader para PIC18.

d) Programador de microcontroladores PIC

Se utiliza una sola vez para grabar el firmware del bootloader en la memoria del microcontrolador.

e) Recursos AN1310

Son indispensables, ya que proporcionan el código fuente de los bootloader para PIC16/PIC18, el software de comunicación con el bootloader desde la computadora, la base de datos para la adición de nuevos microcontroladores y la documentación pertinente.

f) Microchip Solutions

Es una suite de códigos de ejemplo, aplicaciones y librerías de las arquitecturas de microcontroladores de 8 y 16 bits de Microchip. Los bootloaders HID y MCHPFSUSB son proporcionados en esta suite.

Consideraciones sobre la Implementación del Bootloader Serie RS232

Para implementar el bootloader serie RS232 se deben tomar en cuenta las siguientes consideraciones:

1. Bits de configuración del microcontrolador

Cada microcontrolador posee prestaciones diferentes, por lo que los bits de configuración deben configurarse como lo sugiere la tabla 1.

Tabla 1. Bits de configuración sugeridos para el programa bootloader

Categoría	Configuración
Watchdog Timer	"Deshabilitado"
Extended Instruction Set Enable bit	"Deshabilitado"
Bits de configuración del oscilador	De acuerdo al hardware, se recomienda utilizar las frecuencias de operación más altas.
Fail-Safe Clock Monitor Enable bit	"Habilitado", si está disponible
Low-Voltage Program (LVP)	"Deshabilitado", si aplica
Table Read-Protect	"Deshabilitado", si aplica

Estas configuraciones son válidas para microcontroladores PIC16 y PIC18

2. Zona de memoria de almacenamiento del Bootloader

El programa bootloader viene configurado por defecto para guardarse en la parte alta de la memoria de programa del microcontrolador. Sin embargo, se puede configurar para que se guarde en la parte baja. Esto puede resultar útil en microcontroladores PIC18, pero no se recomienda para los PIC16. Si este fuese el caso, se deben quitar los comentarios a la línea 56 del fichero de configuración (header) bootconfig.inc, para indicarle al bootloader que debe cargarse en la zona baja de la memoria de programa.

Al utilizar el bootloader cargado en la zona baja, será necesario que todos los programas a crear especifiquen al programa linker que debe reservar una zona de memoria mediante el comando: codeoffset = 0x400h, esta instrucción reserva 1KB de memoria de programa para el bootloader, evitando que al momento de la programación el firmware sobrescriba al bootloader.

Consideraciones sobre la Implementación del Bootloader HID en PIC18

Aprovechando el puerto USB disponible en microcontroladores PIC18 y superiores, es posible implementar bootloaders tanto HID como MCHPFSUSB, por las ventajas antes mencionadas, es preferible el bootloader HID.

Microchip provee a través de las USB HID Tools, las herramientas necesarias para facilitar la implementación de este tipo de bootloaders. El firmware viene pre compilado para los siguientes microcontroladores: PIC18F14K50, PIC18F4450 y PIC18F4550 escritos en lenguaje C18.

Las siguientes son algunas de las consideraciones más importantes para implementar el bootloader HID:

I. Entrada a modo Bootloader

En el bootloader HID, se requiere que una entrada del microcontrolador sea utilizada para especificar (mediante su activación en bajo) que se debe ingresar al modo bootloader. Por defecto, la entrada utilizada es RB4, aunque se puede cambiar modificando la definición en la línea 105 del archivo de configuración io_cfg.h Para ingresar a modo bootloader, es necesario que esta entrada permanezca en bajo después de un reset del sistema y que la aplicación de comunicación se encuentre en modo de espera.

II. Reserva de memoria de programa.

Por defecto, el bootloader HID, se carga en la zona baja de la memoria de programa. Es necesario entonces, indicarle al programa linker que

nuestros programas se cargarán a partir de la posición 0x1000h por medio del comando codeoffset y extend address.

Conclusión

Experiencias en la implementación de Bootloaders Serie RS232 y HID en ITCA-FEPADE

En la Escuela de Ingeniería Eléctrica y Electrónica de la Escuela Especializada en Ingeniería ITCA-FEPADE, se ha implementado desde el año 2012 el uso de placas de desarrollo con bootloader HID para el PIC18F4550.

Inicialmente, se analizó y estudió la posibilidad de utilizar modelos de placas de desarrollo con bootloader existentes, tanto de código y hardware abierto como propietario. Sin embargo, al analizar las ventajas y desventajas de los productos existentes, se optó por el diseño de un modelo que utilice, en la medida de lo posible, el firmware, herramientas y compiladores que el fabricante del microcontrolador provee de forma gratuita.

Como resultado de este análisis se diseñó la placa de desarrollo que se muestra en la figura 4.

Adicionalmente, se está experimentando con el diseño de placas de desarrollo con bootloaders para microcontroladores PIC16 de bajo costo y alta popularidad con el PIC16F877A y PIC18F887, como se muestra en la figura 5.

Options for xc8 (v1.30)	
Option categories:	Additional options
Reset	
Extra Linker Options	
Serial	...
Codeoffset	0x1000
Checksum	
Errata	
Trace type	(N/A)
Extend address 0 in HEX file	<input checked="" type="checkbox"/>
Use response file to link	<input type="checkbox"/>

Fig. 3. Se debe especificar al linker de XC8 que la aplicación se cargará a partir de la posición de memoria 0x1000h.



Fig. 4. Placa de desarrollo con Bootloader HID utilizando el microcontrolador PIC18F4550 con puerto USB integrado.

La experiencia ha demostrado que cuando los estudiantes y docentes cuentan con una herramienta “portable” para la práctica y experimentación de circuitos microcontrolados, se mejora considerablemente la asimilación de competencias y se facilita notablemente la investigación en áreas afines. Actualmente los estudiantes de módulos y asignaturas relacionadas construyen el modelo de la placa de desarrollo con bootloader y la utilizan en sus prácticas de laboratorios y proyectos de investigación.

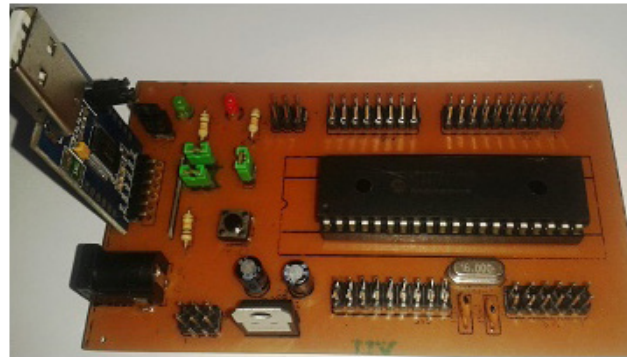


Fig. 5. Vista lateral

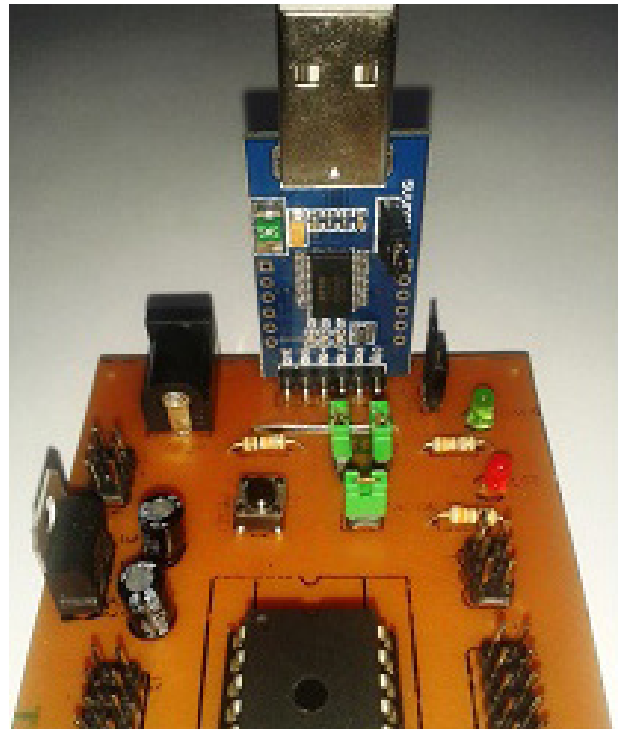


Fig. 5. Placa de desarrollo con Bootloader RS232 utilizando el microcontrolador PIC16F877A. Nótese la utilización de una placa de interfaz FT232RL para brindar conectividad USB con una computadora.

Bibliografía

- SCHLUNDER, E. High-Speed Serial Bootloader for PIC16 and PIC18 Devices [en línea]. Estados Unidos; Microchip Technology, 2010 [fecha de consulta: 19 de junio de 2015]. Disponible: <http://ww1.microchip.com/downloads/en/appnotes/01310a.pdf>
- TAFANERA, Antonio R. Teoría y diseños con microcontroladores PIC. 1a. ed. Argentina: Inca Editorial, 2004. 294 p. ISBN: 9874318686
- VALDÉS Pérez, Fernando E. y PÁLLAS Areny, Ramón. Microcontroladores : fundamentos y aplicaciones con PIC. 1a. ed. México, D. F. : Alfaomega, 2007. 340 p. ISBN: 9789701511497